UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/746,854	12/22/2000	James Morrow	10407/476	7292

30076 7590 10/05/2006

BROWN RAYSMAN MILLSTEIN FELDER & STEINER, LLP
1880 CENTURY PARK EAST
12TH FLOOR
LOS ANGELES, CA 90067

EXAMINER	
PATEL, NIKETA I	

ART UNIT	PAPER NUMBER
2181	

DATE MAILED: 10/05/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/746,854	MORROW ET AL.	
	Examiner	Art Unit	
	Niketa I. Patel	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 August 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-34* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-34* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 April 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 08/16/2006 has been entered.

Double Patenting

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 1-11, 12-18, 19-23, 24-34 are provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1-11, 12-18, 19-23, 24-34,

respectively, of copending Application No. 10/956,431 (hereinafter '*App431*') in view of Montijo U.S. Patent Number: 6,052,107 (hereinafter '*Montijo*'.) This is a provisional obviousness-type double patenting rejection.

4. **Referring to claim 1**, claim 1 of *App431* discloses all of the claimed subject matter including a generic device controller unit system and a method for facilitating interaction between a processor and any number of peripheral devices, however, does not disclose a non-true real time computer having a non-true real time-enabled circuit board. *Montijo* teaches the use of a non-true real time computer having a non-true real time-enabled circuit board [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system], in order to allow a user to enter and/or retrieve various types of data from a peripheral device. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of applicant's invention to provide a non-true real time computer having a non-true real time-enabled circuit board in order to allow a user to enter and/or retrieve data.

5. **Referring to claim 2**, claim 2 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control while interfaced with a non-true real time operating system running standard non-true real time software.

6. **Referring to claim 3**, claim 3 of *App431* discloses wherein the generic device controller unit system functions as a distributed processing environment

7. **Referring to claim 4**, claim 4 of *App431* discloses wherein the generic device controller unit system further includes customized system drivers.

8. **Referring to claim 5**, claim 5 of *App431* discloses wherein Universal Serial Bus is the default communication protocol between the generic device controller unit system and the processor.

9. **Referring to claim 6**, claim 6 of *App431* discloses wherein the generic device controller unit system interfaces with the non-true real time operating system that functions in a Win32 environment.

10. **Referring to claim 7**, claim 7 of *App431* discloses wherein the generic device controller unit system is an input/output device interface for a processor to peripheral devices.

11. **Referring to claim 8**, claim 8 of *App431* discloses wherein the generic device controller unit system provides real time device control to resource management capabilities of a standard non-true real time operating system.

12. **Referring to claim 9**, claim 9 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the higher level functionality of the processor.

13. **Referring to claim 10**, claim 10 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the processor using a true real time kernel.

14. **Referring to claim 11**, claim 11 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the processor utilizing a layered true real time operating system

15. **Referring to claim 12**, claim 12 of *App431* discloses all of the claimed subject matter including a generic device controller unit system and a method for facilitating interaction

between a processor and any number of peripheral devices, however, does not disclose a non-true real time computer having a non-true real time-enabled circuit board. *Montijo* teaches the use of a non-true real time computer having a non-true real time-enabled circuit board [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system], in order to allow a user to enter and/or retrieve various types of data from a peripheral device. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of applicant's invention to provide a non-true real time computer having a non-true real time-enabled circuit board in order to allow a user to enter and/or retrieve data.

16. **Referring to claim 13**, claim 13 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control while interfaced with a non-true real time operating system running standard non-true real time software.

17. **Referring to claim 14**, claim 14 of *App431* discloses wherein the generic device controller unit system functions as a distributed processing environment.

18. **Referring to claim 15**, claim 15 of *App431* discloses wherein the generic device controller unit system is an input/output device interface for a processor to peripheral devices.

19. **Referring to claim 16**, claim 16 of *App431* discloses wherein the generic device controller unit system provides real time device control to resource management capabilities of a standard non-true real time operating system.

20. **Referring to claim 17**, claim 17 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the higher level functionality of the processor.

21. **Referring to claim 18**, claim 18 of *App431* discloses wherein the generic device controller unit system interfaces with the non-true real time operating system that functions in a Win32 environment.

22. **Referring to claim 19**, claim 19 of *App431* discloses all of the claimed subject matter including a generic device controller unit system and a method for facilitating interaction between a processor and any number of peripheral devices, however, does not disclose a non-true real time computer having a non-true real time-enabled circuit board. *Montijo* teaches the use of a non-true real time computer having a non-true real time-enabled circuit board [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system], in order to allow a user to enter and/or retrieve various types of data from a peripheral device. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of applicant's invention to provide a non-true real time computer having a non-true real time-enabled circuit board in order to allow a user to enter and/or retrieve data.

23. **Referring to claim 20**, claim 20 of *App431* discloses wherein the generic device controller unit system functions as a distributed processing environment.

24. **Referring to claim 21**, claim 21 of *App431* discloses wherein Universal Serial Bus is the default communication protocol between the generic device controller unit system and the processor.

25. **Referring to claim 22**, claim 22 of *App431* discloses wherein the generic device controller unit system is an input/output device interface for a processor to peripheral devices.

26. **Referring to claim 23**, claim 23 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the higher level functionality of the processor.

27. **Referring to claim 24**, claim 24 of *App431* discloses all of the claimed subject matter including a generic device controller unit system and a method for facilitating interaction between a processor and any number of peripheral devices, however, does not disclose a non-true real time computer having a non-true real time-enabled circuit board. *Montijo* teaches the use of a non-true real time computer having a non-true real time-enabled circuit board [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system], in order to allow a user to enter and/or retrieve various types of data from a peripheral device. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of applicant's invention to provide a non-true real time computer having a non-true real time-enabled circuit board in order to allow a user to enter and/or retrieve data.

28. **Referring to claim 25**, claim 25 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control while interfaced with a non-true real time operating system running standard non-true real time software.

29. **Referring to claim 26**, claim 26 of *App431* discloses wherein the generic device controller unit system functions as a distributed processing environment.

30. **Referring to claim 27**, claim 27 of *App431* discloses wherein the generic device controller unit system further includes customized system drivers.

31. **Referring to claim 28**, claim 28 of *App431* discloses wherein Universal Serial Bus is the default communication protocol between the generic device controller unit system and the processor.

32. **Referring to claim 29**, claim 29 of *App431* discloses wherein the generic device controller unit system interfaces with the non-true real time operating system that functions in a Win32 environment.

33. **Referring to claim 30**, claim 30 of *App431* discloses wherein the generic device controller unit system is an input/output device interface for a processor to peripheral devices.

34. **Referring to claim 31**, claim 31 of *App431* discloses wherein the generic device controller unit system provides real time device control to resource management capabilities of a standard non-true real time operating system.

35. **Referring to claim 32**, claim 32 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the higher level functionality of the processor.

36. **Referring to claim 33**, claim 33 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the processor using a true real time kernel.

37. **Referring to claim 34**, claim 34 of *App431* discloses wherein the generic device controller unit system produces true real time peripheral device control without the processor utilizing a layered true real time operating system.

Claim Objections

38. Claims 12-34 are objected to because of the following:

a. **Claim 12** recites a limitation of 'a processor' at line 2 and at line 7. It is unclear as to whether the applicant is introducing, at line 7, another processor separate and distinct form the processor claimed in line 2 or due to typographical error, recited the limitation of 'a processor' rather than reciting '*the processor*'. It appears as though the applicant has inadvertently recited 'a processor' when in fact the intention was to refer back to the earlier claimed processor by reciting '*the processor*' at line 7 of the claim. The remainder of the Office Action will be written as if this were the case.

The Appropriate correction is required.

Claims 13-18 depend on claim 12 and therefore inherit the same deficiency.

b. **Claim 19** recites a limitation of 'a processor' at line 2 and at line 5. It is unclear as to whether the applicant is introducing, at line 5, another processor separate and distinct form the processor claimed in line 2 or due to typographical error, recited the limitation of 'a processor' rather than reciting '*the processor*'. It appears as though the applicant has inadvertently recited 'a processor' when in fact the intention was to refer back to the earlier claimed processor by reciting '*the processor*' at line 5 of the claim. The remainder of the Office Action will be written as if this were the case.

Furthermore, claim 19 recites a limitation of 'any number of peripheral devices' at lines 2-3 and recites 'various peripheral devices' at line 5. It is unclear as to whether the applicant is introducing, at line 5, another set of peripheral devices separate and distinct form the earlier claimed peripheral devise or due to typographical error, recited

the limitation of 'various peripheral devices' rather than reciting '*said any number of peripheral devices*'. It appears as though the applicant has inadvertently recited 'various peripheral devices' when in fact the intention was to refer back to the earlier claimed peripheral devices by reciting '*said any number of peripheral devices*' at line 5. The remainder of the Office Action will be written as if this were the case.

The Appropriate correction is required. Claims 20-23 depend on claim 19 and therefore inherit the same deficiency.

c. **Claim 24** recites a limitation of 'any number of peripheral devices' at lines 2-3 and recites 'various peripheral devices' at line 4. It is unclear as to whether the applicant is introducing, at line 4, another set of peripheral devices separate and distinct form the earlier claimed peripheral devise or due to typographical error, recited the limitation of 'various peripheral devices' rather than reciting '*said any number of peripheral devices*' in order to refer back to the earlier claimed peripheral devices. It's further confusing when the claim recites 'the peripheral devices' at line 11, is the intention, at line 11 to refer back to the 'various peripheral devices' or 'any number of peripheral devices', in the event that they are different set of peripheral devices? It appears as though the applicant has inadvertently recited 'various peripheral devices' when in fact the intention was to refer back to the earlier claimed peripheral devices by reciting '*said any number of peripheral devices*' at line 3. The remainder of the Office Action will be written as if this were the case.

The Appropriate correction is required. Claims 25-34 depend on claim 24 and therefore inherit the same deficiency.

Claim Rejections - 35 USC § 112

39. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter, which the applicant regards as his invention.

40. Claims 1-34 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

d. Claims 1, 12, 19 and 24, recite the limitation of 'a processor' and 'a non-true real time computer'. It is unclear as to which processor and the computer the applicant is referring to since, the specification describes that the processor (element 40) of figure 2 can be a computer (specification page 9, lines 15-17) and the generic device controller unit (GDCU) (figure 1) can be a computer as well (see page 11, lines 20-26 and page 14, lines 8-9.) It appears as though the applicant has inadvertently used two different aliases ('a processor' and 'a non-true real time computer') to refer to the same element 40 of the figure 2. The remainder of the Office Action will be written as if this were the case.

e. Claims 2-11, 13-18, 20-23 and 25-34 depend on claims 1, 12, 19 and 24 respectively, and therefore inherit the same deficiency.

Claim Rejections - 35 USC § 103

41. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are

> such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

42. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

43. Claims 1-4, 6-20, 22-27 and 29-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Swales et al. U.S. Patent Number: 6,233,626 B1 (hereinafter '*Swales*') and further in view of *Montijo*.

44. **Referring to claim 1**, *Swales* teaches a generic device controller unit system [see figure 3] for facilitating interaction between a processor [see figure 3, element 12, 'Master Device'] and any number of peripheral devices [see figure 3, element 14, 'I/O Device' and column 1, lines 23-42, 'field devices'], the system comprising: a general purpose device controller [see figure 3, element 10 and column 4, lines 5-7, 'the COM-adapter'] employing asynchronous true real time peripheral device control [see column 1, lines 23-42 and column 4, sentence beginning at line 66, field devise are controlled via specific true real time control protocols, such as MODBUS plus, Interbus-S, Profibus DP, Echelon, Seriplex, CAN DeviceNet, CAN SDS], wherein the device controller interfaces between the peripheral devices and a non-true real time computer a non-true real time operating system [see figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/

field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running on non-true real time operating system (i.e., Windows 95 & NT)], thereby allowing a non-true real time operating system to implement true real time control of the peripheral devices [see column 3, lines 15-23, 46-53, COM-adapter allows the field master (i.e., the host device) to control field devices in real time without special operating system]; and a data and protocol communications interface [see figure 3, elements 22, 'ATI interface'], wherein the communications interface connects the processor and the peripheral devices [see figure 3, elements 22 and ATI interface are part of element 10 which provides communication interface between the host device (element 12, master device) and the peripheral devices (element 14, I/O device)], thereby allowing the processor to utilize a single protocol and associated data to communicate with the peripheral devices which may be utilizing protocols and associated data which are different than that used by the processor [see column 6, lines 20-25, the host side uses TCP/IP protocol which is different then the protocol used by the field devices, ATI protocol, as described in column 9, lines 30-34.]

Swales does not set forth the detailed limitation of the non-true real time computer having an non-true real time-enabled circuit board however, *Montijo* teaches a non-true real time computer having a non-true real time operating system and non-true real time-enabled circuit board [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system] in order to allow the host computer to process information.

One of ordinary skill in the art at the time of applicant's invention would have clearly recognized that it is quite advantageous for the non-true real time computer of *Swales* to have a

non-true real time operating system and non-true real time-enabled circuit board to provide the host computer with information processing capability. It is for this reason that one of ordinary skill in the art would have been motivated to implement the non-true real time computer of *Swales* with a non-true real time operating system and non-true real time-enabled circuit board in order to allow the host computer to process information.

45. **Referring to claim 12**, *Swales* teaches a generic device controller unit system [see figure 3] for facilitation interaction between a processor [see figure 3, element 12, 'Master Device'] and any number of peripheral devices [see figure 3, element 14, 'I/O Device' and column 1, lines 23-42, 'field devices'], the system comprising: a general purpose device controller [see figure 3, element 10 and column 4, lines 5-7, 'the COM-adapter'] employing asynchronous true real time peripheral device control [see column 1, lines 23-42 and column 4, sentence beginning at line 66, field devise are controlled via specific true real time control protocols, such as MODBUS plus, Interbus-S, Profibus DP, Echelon, Seriplex, CAN DeviceNet, CAN SDS], wherein the device controller interfaces between the peripheral device and a non-true real time computer having a non-true real time operating system [see figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running a non-true real time operating system (i.e., Windows 95 & NT)] thereby allowing a non-true real time operating system to implement true real time control of the peripheral devices without a processor requiring either a real time kernel or a layered true real time operating system [see column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a

host running Windows OS to control field devices which are using real time protocol such as MODBUS.]

Swales does not set forth the detailed limitation of the non-true real time computer having an non-true real time-enabled circuit board however, *Montijo* teaches a non-true real time computer having a non-true real time operating system and non-true real time-enabled circuit board and [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system] in order to allow the host computer to process information.

One of ordinary skill in the art at the time of applicant's invention would have clearly recognized that it is quite advantageous for the non-true real time computer of *Swales* to have a non-true real time operating system and non-true real time-enabled circuit board to provide the host computer with information processing capability. It is for this reason that one of ordinary skill in the art would have been motivated to implement the non-true real time computer of *Swales* with a non-true real time operating system and non-true real time-enabled circuit board in order to allow the host computer to process information.

46. **Referring to claim 19**, *Swales* teaches a generic device controller unit system [see figure 3] for providing a data and protocol communication interface which facilitates interaction between a processor [see figure 3, element 12, 'Master Device'] and any number of peripheral devices [see figure 3, element 14, 'I/O Device' and column 1, lines 23-42, 'field devices'], the system comprising: an asynchronous general device data and protocol communications interface [see figure 3, element 10 and column 4, lines 5-7, 'the COM-adapter'], wherein the communications interface connects a processor and various peripheral devise [see figure 3,

element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master], thereby allowing the processor to unitize a single protocol and associated data to communicate with the various peripheral devices which may utilize different protocols and associated data then that used by the processor [see column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS] and wherein the communications interface employs asynchronous true real time peripheral device control [see column 1, lines 23-42 and column 4, sentence beginning at line 66, field devise are controlled via specific true real time control protocols, such as MODBUS plus, Interbus-S, Profibus DP, Echelon, Seriplex, CAN DeviceNet, CAN SDS], and wherein the communications interface connects the peripheral devices and a non-true real time computer having a non-true real time operating system [see figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running a non-true real time operating system (i.e., Windows 95 & NT).]

Swales does not set forth the detailed limitation of the non-true real time computer having an non-true real time-enabled circuit board however, *Montijo* teaches a non-true real time computer having a non-true real time operating system and non-true real time-enabled circuit board and [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system] in order to allow the host computer to process information.

One of ordinary skill in the art at the time of applicant's invention would have clearly recognized that it is quite advantageous for the non-true real time computer of *Swales* to have a non-true real time operating system and non-true real time-enabled circuit board to provide the host computer with information processing capability. It is for this reason that one of ordinary skill in the art would have been motivated to implement the non-true real time computer of *Swales* with a non-true real time operating system and non-true real time-enabled circuit board in order to allow the host computer to process information.

47. **Referring to claim 24**, *Swales* teaches a method for providing a data and protocol communications interface to facilitate interaction between a processor [see figure 3, element 12, 'Master Device'] and any number of peripheral devices [see figure 3, element 14, 'I/O Device' and column 1, lines 23-42, 'field devices'], the method comprising: interfacing between various non-specific peripheral devices [see figure 3, element 14, 'I/O Device' and column 1, lines 23-42, 'field devices'] a non-true real time computer having a non-true real time operating system [see figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running a non-true real time operating system (i.e., Windows 95 & NT)]; employing asynchronous true real time peripheral device control through a generic device controller unit [see figure 3, element 10 and column 4, lines 5-7, 'the COM-adapter'], wherein the device controller allows the processor to implement true real time control of the peripheral devices without the non-true real time operating system requiring either a real time kernel or a layered true real time operating system [see column 3, lines 11-23, 46-53, column 6, lines 14-28

and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS]; and providing a protocol and associated data communications interface [see figure 3, elements 22, 'ATI interface'] between the processor and the peripheral devices [see figure 3, elements 22 and ATI interface are part of element 10 which provides communication interface between the host device (element 12, master device) and the peripheral devices (element 14, I/O device)], thereby allowing the processor to utilize a single protocol and associated data to communicate with the peripheral devices which may utilize different protocols and associated data then that used by the processor [see column 6, lines 20-25, the host side uses TCP/IP protocol which is different then the protocol used by the field devices, ATI protocol, as described in column 9, lines 30-34.]

Swales does not set forth the detailed limitation of the non-true real time computer having an non-true real time-enabled circuit board however, *Montijo* teaches a non-true real time computer having a non-true real time operating system and non-true real time-enabled circuit board and [see *Montijo* column 4, lines 31-49, 'a computer motherboard' and lines 62-67, 'operating system' and column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., the non-true real time operating system] in order to allow the host computer to process information.

One of ordinary skill in the art at the time of applicant's invention would have clearly recognized that it is quite advantageous for the non-true real time computer of *Swales* to have a non-true real time operating system and non-true real time-enabled circuit board to provide the host computer with information processing capability. It is for this reason that one of ordinary skill in the art would have been motivated to implement the non-true real time computer of

Swales with a non-true real time operating system and non-true real time-enabled circuit board in order to allow the host computer to process information.

48. **Referring to claims 2, 13, 25**, combination of *Swales & Montijo* teaches wherein the generic device controller unit system produces true real time peripheral device control while interfaced with a non-true real time operating system running standard non-true real time software [see *Swales* figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running a non-true real time operating system (i.e., Windows 95 & NT).]

49. **Referring to claims 3, 14, 20, 26**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system functions as a distributed processing environment [see column 6, lines 14-23 and column 3, lines 15-23, 46-53, networked devices COM-adapter, host computer and field devices provide distributed processing.]

50. **Referring to claims 4, 27**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system further includes customized system drivers [see *Swales* column 6, lines 42-63, kernel firmware of the COM-adapter.]

51. **Referring to claims 6, 18, 29**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system interfaces with the non-true real time operating system that functions in a Win32 environment **[see** *Swales* column 5, lines 1-6, 'Windows 95, 3.1, NT operating system' i.e., Win32 environment non-true real time operating system.]

52. **Referring to claims 7, 15, 22, 30**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system is an input/output device interface for a processor to peripheral devices [see *Swales* figure 3, element 10 and column 3, lines 46-53 and column 4, lines 5-7, the COM-adapter provides communication interface to a master processor and an I/O device.]

53. **Referring to claims 8, 16, 31**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system provides real time device control to resource management capabilities of a standard non-true real time operating system [see *Swales* column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS, the host does runs Windows operating system, Windows operating system do not run true real time kernel; also see column 3, 15-23, which specifically discloses that real time component are not required, instead standard network components are shared.]

54. **Referring to claims 9, 17, 23, 32**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system produces true real time peripheral device control without the higher level functionality of the processor [see *Swales* column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS, the host does runs Windows operating system, Windows operating system do not run true real time kernel; also see column 3, 15-23, which specifically discloses that real time component are not required, instead standard network components are shared.]

55. **Referring to claims 10, 33**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system produces true real time peripheral device control without the processor using a true real time kernel [see *Swales* column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS, the host does runs Windows operating system, Windows operating system do not run true real time kernel; also see column 3, 15-23, which specifically discloses that real time component are not required, instead standard network components are shared.]

56. **Referring to claims 11, 34**, combination of *Swales & Montijo* teaches the system and the method wherein the generic device controller unit system produces true real time peripheral device control without the processor utilizing a layered true real time operating system [see *Swales* column 3, lines 11-23, 46-53, column 6, lines 14-28 and column 1, lines 23-42, the COM-adapter allows a host running Windows OS to control field devices which are using real time protocol such as MODBUS; also see column 3, 15-23, which specifically discloses that real time component are not required, instead standard network components are shared.]

57. As far as the examiner can interpret the claims in light of the 35 U.S.C. 112, second paragraph, rejection supra, claims 5, 21, 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Swales & Montijo* as modified above in claims 1, 19, 24 and further in view of Evoy et al. U.S. Patent Number: 5,958,020 (hereinafter "*Evoy*".)

58. **Referring to claims 5, 21, 28**, combination of *Swales & Montijo* teaches a generic device controller unit system and a method for facilitating interaction between a processor and any

number of peripheral devices [*Swales* figure 3, element 10 located between elements 12 & 14 and column 1, lines 64-67 and column 2, lines 1-5, a personal computer/ host computer/ host device/ field master; column 6, lines 14-28 states that the COM-adapter is compatible with host program running over Windows 95 & NT, therefore the host is running a non-true real time operating system (i.e., Windows 95 & NT) and *Montijo* teaches serial port, figure 6, element 610.] The combination of *Swales & Montijo* does not set forth the limitation wherein Universal Serial Bus is the default communication protocol between the generic device controller unit system and the processor, however *Evoy* teaches a use of Universal Serial Bus protocol between the generic device controller unit system and the processor [see *Evoy* column 2, lines 46-60 and column 1, lines 23-43] because USB connects peripheral devices to the resources of the computer system without consuming the input output resources of the computer system and also provides for automatic USB peripheral device configuration and eliminates computer system resource conflicts.

One of ordinary skill in the art at the time of applicant's invention would have clearly recognized that it is quite advantageous for the system of *Swales & Montijo* to be able to automatically configure peripheral devices in order to eliminate computer system resource conflicts by using USB protocol. It is for this reason that one of ordinary skill in the art would have been motivated to implement USB protocol in the system of *Swales & Montijo* to eliminate computer system resource conflicts and saving input output resources.

Response to Arguments

59. Applicant's arguments with respect to claim 1-34 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

60. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

f. **Definition of MODBUS** (From Wikipedia): is serial communication protocol published by Modicon in 1979 for use with its programmable logic controllers (PLCs). It has become a de facto standard communication protocol in industry, and is now the most commonly available means of connecting industrial electronic devices. Modbus allows for communication between many devices connected to the same network, for example a system that measures temperature and humidity and communicates the results to a supervisory computer.

g. **Definition of Distributed Processing/Computing** (From Wikipedia): is a programming paradigm focusing on designing distributed, open, scalable, transparent, fault tolerant systems. This paradigm is a natural result of the use of computer to form networks. Distributed computing is decentralized and parallel computing, using two or more computers communicating over a network to accomplish a common objective or task.

h. The following documents have been made record of to further show the state of the art as it pertains to a use of generic industrial controller to allow a persona computer to control industrial/field devices without having real time operating system:

Dummermuth U.S. Patent Number: 6,009,454

Schmidt et al. U.S. Patent Number: 6,018,797

Almstead et al. U.S. Patent Number: 6,499,114 B1

Bowling U.S. Patent Number: 5,752,008

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Niketa I. Patel whose telephone number is (571) 272 4156. The examiner can normally be reached on M-F 8:00 A.M. to 5:00 P.M.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Fritz Fleming can be reached on (571) 272 4145. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Examiner: Niketa Patel
09/27/2006